

开发月刊

Development Monthly

2012年04月
总第013期

HTML



关于HTML5的七个传说

MARK ONLY

Java优势依然 甲骨文更爆十年规划

开发者应该关注的五项Web新兴技术



ONE COLOR LOGOS

SVG
Dark White
PNG
Dark White



STICKER TEMPLATES

SVG
Download
PNG
Standard Size

developer.51cto.com



	编程排行 Billboard
3	4月编程语言排行榜:C语言荣归宝座
	专题报道 《HTML 5开发》
5	关于HTML 5的七个传说
7	Visual Studio 11针对HTML 5的改进
9	HTML 5数据可视化与科学研究
	技术热点 Technology hot
11	宅男程序员给老婆课程9：数据模型
13	为什么我不再做.NET开发
14	传统程序员的出路在何方？
16	Centos5下MySQL Proxy0.8.2安装
19	IBM副总裁Judy:下代DB2年内发布
20	从个人网站到淘宝网 仰观Java时代淘宝的技术发展
23	面向对象的JavaScript基本知识指南
25	最喜欢与最讨厌的编程语言
26	Java优势依然 甲骨文更爆十年规划
28	开发者应该关注的五项Web新兴技术
30	编程语言新生儿到底因何成功 又为何失败？

4月编程语言排行榜:C语言荣归宝座

TIOBE 近日公布了 2012 年 4 月份的编程语言排行榜,终于不出小编所料,在上个月的编程语言排行榜中说过的“编程语言的王者之争不久很可能会发生改变”实现了,一方面是 Java 在上几个月中一直属于下滑状态,另一方面是 C 这具有 30 多年历史的编程语言依然是历久弥新,一些优秀特性至今一直都被广泛应用。

TIOBE 近日公布了 2012 年 4 月份的编程语言排行榜,终于不出小编所料,在上个月的编程语言排行榜中说过的“编程语言的王者之争不久很可能会发生改变”实现了。

一方面是 Java 在上几个月中一直属于下滑状态,另一方面是 C 这具有 30 多年历史的编程语言依然是历久弥新,一些优秀特性至今一直都被广泛应用。

C 语言从 2010 年 6 月份丢失了宝座后一直都是在 Java 后面穷追不舍,终于在本月荣归了宝座。

当然了, C 和 Java 这两们语言所占的份额相差不多,也会经常来回波动,相信 Android 平台的普及, Java 份额的下降也不会维持多久,不过未来的几个月内 C 还是会保持领先的位置。

值得我们关注的还有 Objective-C,随着 The New PAD 的发布也一路的飙升,超越了 C#, 占据了第 4 的位置。

而 C++ 也重新回来了第 3 的位置。浮动最大的就是 T-SQL 了,从原来的 14 位置滑到了 18,份额也从原来的 0.925% 下降到了 0.760%。

下面是本期编程语言排行榜的其他排名数据和趋势走向

下面是前 20 名的编程语言排行

Position Apr 2012	Position Apr 2011	Delta in Position	Programming Language	Ratings Apr 2012	Delta Apr 2011	Status
1	2	↑	C	17.565%	+1.30%	A
2	1	↓	Java	17.026%	-2.02%	A
3	3	=	C++	8.036%	-0.01%	A
4	8	↑↑↑↑	Objective-C	8.235%	+0.05%	A
5	4	↓	C#	7.348%	+0.16%	A
6	5	↓	PHP	5.288%	-1.30%	A
7	7	=	(Visual) Basic	4.862%	+0.28%	A
8	6	↓↓	Python	3.866%	-1.27%	A
9	10	↑	JavaScript	2.879%	+1.37%	A
10	9	↓	Perl	2.017%	+0.40%	A
11	11	=	Ruby	1.510%	+0.01%	A
12	24	↑↑↑↑↑↑↑	PL/SQL	1.073%	+0.92%	A
13	13	=	Delphi/Object Pascal	1.070%	+0.34%	A
14	15	↑↑↑↑↑	Visual Basic .NET	0.970%	+0.64%	A
15	15	=	Lisp	0.851%	+0.02%	A
16	17	↑	Pascal	0.812%	+0.10%	A
17	18	↓	Ada	0.783%	+0.01%	A-
18	18	=	Transact-SQL	0.760%	+0.18%	A
19	22	↑↑↑	Logo	0.502%	+0.12%	B
20	52	↑↑↑↑↑↑↑↑	NXT-G	0.578%	+0.35%	B

C 语言简介

C 语言是一种计算机程序设计语言。它既具有高级语言的特点,又具有汇编语言的特点。它由美国贝尔研究所的 D.M.Ritchie 于 1972 年推出。1978 后, C 语言已先后被移植到大、中、小及微型机上。它可以作为工作系统设计语言,编写系统应用程序,也可以作为应用程序设计语言,编写不依赖计算机硬件的应用程序。它的应用范围广泛,具备很强的数据处理能力,不仅仅是在软件开发上,而且各类科研都需要用到 C 语言,适于编写系统软件,三维,二维图形和动画。

本文未完,请阅读原文: ■

<http://developer.51cto.com/art/201204/327422.htm>

Aloha Editor makes CONTENT in your browser EDITABLE



HTML 5 是近十年来 Web 开发标准最巨大的飞跃。和以前的版本不同，HTML 5 并非仅仅用来表示 Web 内容，它的新使命是将 Web 带入一个成熟的应用平台，在 HTML 5 平台上，视频，音频，图象，动画，以及同电脑的交互都被标准化。本文专题将带您体验 HTML 5，深入下一代 Web 开发标准。

HTML 5 简单的说就是一组新的标准，语法和规则用来创建网页，可以利用一些新的浏览器的“本地”功能，例如离线存储，多媒体播放和其他的交互等等。我们所看到的很多很酷的 HTML 5 网站实际上跟 HTML 5 无关，而是由 CSS3 和 jQuery 生成的。CSS3 和 jQuery 是和 HTML 5 一样的新技术，它们和 HTML 5 是平行的概念。

从浏览器的角度来说，是的，添加了 HTML 5 doctype 的网站都会被认为是 HTML5 网站。但是 HTML 5 不仅仅是一个 doctype。它是一套完整的最佳实践、语义标签、排版元素例如 <header>，<nav>，<section>，<article> 等等。这些元素能让你的网站利用浏览器的新功能，才能称得上是 HTML 5。同时你也要注意那些在 HTML5 中被弃用的元素和属性，以保证兼容性。

熟练的掌握 HTML 5 并不会让你看起来更帅，除非你是 Bruce Lawson。它也不会让你找到梦想的工作。你的实力还是需要通过设计能力，编码能力来体现。当然，学习新技术的热情和毅力是值得欣赏的。你有可能获得同事的尊重，如果你用你的所学帮助他们的话。

51CTO 开发频道语

关于HTML 5的七个传说

HTML 5 简单的说就是一组新的标准,语法和规则用来创建网页,可以利用一些新的浏览器的“本地”功能,例如离线存储,多媒体播放和其他的交互等等。我们所看到的很多很酷的 HTML5 网站实际上跟 HTML 5 无关,而是由 CSS 3 和 jQuery 生成的。CSS3 和 jQuery 是和 HTML5 一样的新技术,它们和 HTML5 是平行的概念。

这篇文章讲了关于 HTML5 的 7 个传说,也就是误解:)

传说 #1: 因为苹果不支持 Flash,所以人们发明了 HTML5

树大招风,苹果似乎经常受到各种指责。虽然 HTML5 可以为移动设备提供一些非常有用的功能,但是它不是为了这个需求而被创造的。事实上,早在 2004 年,Opera 和 Mozilla 为了解决乱糟糟的网页标准定义了一些好的设计标准。他们的目的很简单,就是向 W3C 提交这些标准,从而创建一个更好用,更快速的 HTML 标准。这个标准直到 2006 年才被接受。而在短短的三年内,人们又起草了一个新的标准,审议通过并开始使用。

传说 #2: HTML5 要到 2022 年才会成熟

我不知道这句话是谁传出来的。我们知道 HTML5 的起草文件已经在 2009 年审议通过,也就是没有新的内容可以被添加,也不会有显著的改动。你现在就可以使用 HTML5 了,并且不用担心向下兼容问题。你可以通过一些轻量

级的 JavaScript 来让老版本的浏览器支持复杂的 Canvas。所有新版本的浏览器支持 HTML5。你需要担心的不是 HTML 标签,而是 CSS。很多老版本的浏览器对 CSS3 的支持很差。

传说 #3: HTML5 需要 CSS3 支持

虽然 CSS3 能为 HTML5 增色不少,例如语义 class 名字,但这决不是必须的。HTML5 只是标记语言,你当然可以使用 HTML5 + CSS2,只不过没有那么优雅。由于 HTML5 移除了属性和标签,所以你需要为每一个元素指定全部的样式。例如,在 HTML5 中,为了让浏览器正确的渲染页面,你需要为每一个页面元素设置“display:block”声明。简而言之,HTML5 需要样式来让页面更好看,但不必须是 CSS3。

传说 #4: HTML5 破坏了可访问性 (Accessibility)

相反,HTML5 从设计之初就考虑到了可访问性。每一个元素都支持 WIA ARIA 标志角色。这些角色可以让屏幕阅读器这类设备更好的理解页面的内容。

传说 #5: HTML5 是 Flash 杀手

Flash 依然被无数的网站和开发者使用。关于 Flash 和 HTML5 之间的误解来源于 Adobe 和 Apple 之间的争执以及 HTML5 音频和视频应用的迅速普及。虽然 HTML5 可以很好的支持少量的视频播放,但是 HTML5 比 Flash 需要更多的编码资源,而且缺少那些浏览器插件提供的技术。目前类似 Vimeo 和 YouTube 大量的使用

关于 HTML 5 的七个传说 II

HTML5 技术是为了推广这项技术,他们短期内不会彻底放弃 Flash。开发者 Viki Hoo 关于 Flash 仍然会广泛的被使用有一些令人信服的观点。

但是从页面设计的角度来看,我们希望尽快的从 Flash 中解脱出来。HTML5 学起来用起来都更容易。它不需要插件,装载快速,并且对移动设备很友好。所以不是 HTML5 干掉了 Flash,准确的说是 HTML5 将统治移动设计和开发市场。

传说 #6: 只要在我的网站上添加了 HTML5 doctype,它就是 HTML5 网站了

从浏览器的角度来说,是的,添加了 HTML5 doctype 的网站都会被认为是 HTML5 网站。但是 HTML5 不仅仅是一个 doctype。它是一套完整的最佳实践、语义标签、排版元素例如 <header>, <nav>, <section>, <article> 等等。这些元素能让你的网站利用浏览器的新功能,才能称得上是 HTML5。同时你也要注意那些在 HTML5 中被弃用的元素和属性,以保证兼容性。

传说 #7: 使用 HTML5 会让我获得更多的工作 / 金钱 / 尊重……

熟练的掌握 HTML5 并不会让你看起来更帅,除非你是 Bruce Lawson。它也不会让你找到梦想的工作。你的实力还是需要通过设计能力,编码能力来体现。当然,学习新技术的热情和毅力是值得欣赏的。你有可能获得同事的尊重,如果你用你的所学帮助他们的话。

下面是 HTML 5 APIs 程序员指南

有关 HTML5 中的语法元素相信大家已经看到过不少介绍了,但是关于 HTML5 的 APIs 的介

绍就不那么常见了,而 HTML5 规范主要就是由这些 APIs 构成的。

HTML5 的标准有两个版本,一个是由 W3C 发布的,另一个则是由 WHATWG 发布的。WHATWG 现有的 HTML5 标准中补充了一些 W3C 标准中没有的 APIs (这些 APIs 其实 W3C 也有涉及,只是被单独列在一个补充标准中了)。

除了规范中介绍的 APIs 以外,还有一些相关的 APIs 也遵循了 HTML5 的标准,并也声称自己是 HTML5 的 API。有些 APIs 已经被采用了一段时间了,但是仍然没有被明确列入规范之中,而有些 HTML5 则着手准备对其进行改进。

这篇文章并不准备从代码的角度去介绍这些 APIs,而是会分析他们的用途以及进展情况。还会给出一些参考信息方便读者更加深入地了解自己感兴趣的内容。

HTML5 标准中的 APIs

首先看看 W3C 的 HTML5 规范中的 APIs。

多媒体 API

多媒体 API 是 HTML5 的多媒体元素中的一部分,其中包括 视频 和 音频 元素。这些元素本身实现起来并不复杂,但是这些 APIs 可以使用 JavaScript 方法,这是很多人不太了解的。其中包括 play()、pause()、load() 以及 canPlayType() 这些方法。许多方法不仅可以使使用多媒体类型属性,还可以使用许多其他的属性(例如 poster 属性),这些方法在视频元素中尤其常见。通过在 API 中使用合适的事件和属性,开发者能够实现多媒体的自定义控制。■更多内容请查看

<http://developer.51cto.com/art/201202/314266.htm>

Visual Studio 11 针对 HTML 5 的改进

Mads Kristensen 在 TechDays 2012 海牙站上发表了一场很有趣的演讲,演讲内容是关于在 Visual Studio 11 和 ASP.NET 4.5 中所有的新功能和新选项。



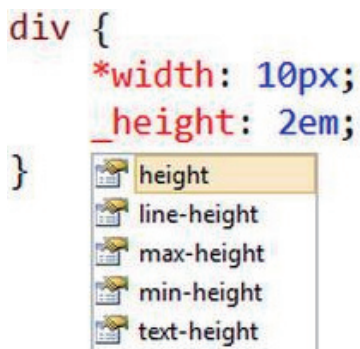
Mads Kristensen 在 TechDays 2012 海牙站上发表了一场很有趣的演讲,演讲内容是关于在 Visual Studio 11 和 ASP.NET 4.5 中所有的新功能和新选项。

然而,当我们浏览这些新功能列表时,我们获得的印象是,这一次的主要焦点在 web 标准和 HTML5 上。

为了使 HTML5、CSS 3 和 Javascript 能够在 Visual Studio 环境中相当容易的被使用,微软做了很多工作。Javascript 和 CSS 编辑器都是全新构建的。因此,它们能够处理所有 Javascript 和 CSS 最新版本提供的新功能。现在,让我们看看 Visual Studio 11 为支持构建适应最新 web 标准和 HTML5 的应用提供了哪些新东西吧。

自动 CSS Hacks

我们都知道 HTML5 和 CSS3 效果很酷而且简单易行。



但是,当创建一个更复杂的 web 应用程序时,常用的方法是我们需要针对特定的浏览器做

出与之相应的设置(这一过程我们把它叫做 CSS Hacks)。而这一步骤可能会花去很多时间并且结果令人沮丧。Visual Studio 11 通过智能提示以及 ‘*’ 和 ‘_’ hacks 属性验证来支持 CSS Hacks。

CSS 的注释和反注释规则

在 CSS 编辑器里写代码时,有多少次我们为不能用快捷键 CTRL+K,C 去注释和用 CTRL+K,U 去反注释而感到沮丧呢?

这些指令为什么我们可以在其他开发环境中使用,却不可能在 CSS 编辑器里使用呢?幸运的是,这种情况已经得到改观,而且,从现在开始,我们不仅可以在代码编辑器里使用这些快捷键,在 CSS 编辑器里同样可以使用。

```
body
{
  -ms-transition-timing-function: ease-in-out;
  -moz-transition-timing-function: ease-in-out;
  -o-transition-timing-function: ease-in-out;
  -webkit-transition-timing-function: ease-in-out;
  transition-timing-function: ease-in-out;
}
```

CSS 编辑器里的代码段

代码分段是 Visual Studio 一个强大的工作机制。唯一的问题是它只能被用在代码编辑器里, CSS 编辑器则不具备这一功能。但在新的 VS 的集成开发环境里,这一状况已经被改善。

现在,许多需要一些特殊浏览器设置的 CSS3 属性都成为准备片段的一部分。

刷新网站：用 Ajax 构建更好的单页面体验 II

拾色器

新的 CSS 编辑器的功能已经被加强,你现在可以通过智能提示来选择一种在你的样式表里已经存在的颜色,或者干脆是通过拾色器控件来选择其它颜色。此功能在 Scott Guthrie 的一篇博客文章里有详细的说明。

Javascript 编辑器

在 Visual Studio 里用 Javascript 最大的问题是它运行起来通常比较慢(尤其是当你正在调试的时候)。而新版本的 Javascript 编辑器是基于 Chakra 引擎的,这将使运行加快。

此外,VS 的 Javascript 编辑器还有其它一些额外的改进,以确保它对于开发者来说能力更强大,更具有可用性。通过添加一些在代码编辑器和其它新工具里可用的选项,这一目标已大部分被实现。下面,我罗列出一些最有趣的选项:

◆ “转到定义”选项。你现在在 Javascript 的文件里也可以使用这一选项,从而可以更快的从一个函数跳转到另一个函数。

◆ 大括号的匹配。你是否经常遇到一些由于括号不完整而造成 Javascript 抛出错误的问题?然后,你是不是不得不一个接一个的寻找那个被遗漏的括号呢?在这款新的 Javascript 编辑器里,这个问题将不会出现。“括号匹配”这样一个在代码编辑器里表现出色的功能也将被添加到 Javascript 编辑器里。它以这样的方式来运行——当你把指针放在其中一个括号上时,它会自动地高亮显示和它匹配的那一个括号。

◆ 隐式引用。现在你可以添加某一个 JS 文件到中心文件列表里,而没有必要去把它添加到每一个要使用它的地方。更好的是,当一个 Javascript 文件被添加到中心文件列表里时,它会生成一个智能提示以表示这个文件对于其他任何一个 Javascript 代码块或文件都是可用的。

◆ 支持 ECMAScript5。ECMAScript5 是最新版本的标准,此标准用以描述 Javascript, JScript 和 ActionScript 语言。新的 Javascript 编辑器支持 ECMAScript5 中新的 syntax 和 APIs。

革命或只是进化?

读完这篇文章后,你可以得出这样一个结论:VS11 包含了对 CSS3、HTML5 和 Javascript 的相当轻微的改善。真正的革命没有发生。可是另一方面,当你把上述所有的变化加起来后,你会发现整个 VS 开发环境的可用性有了一个相当大的改善。而这些改善,将更有利于前端开发。

开发小 TIP

CSS 设计小技巧

border 也就是边框的值,我在做项目开发的时候,很习惯的做法是:

border:1px solid red; 第一个指的是框的大小,第二个是指样式,第三个是指颜色。不过,这里 3 个并非都是必须用的,必须要用的是 border 的样式。其余 2 个有默认值。

比如,你这样写:

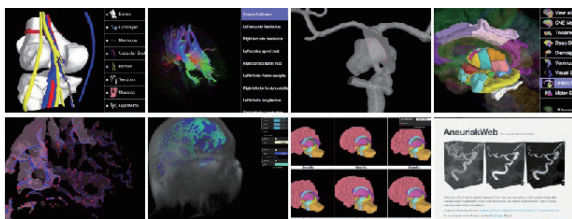
border: solid

HTML 5 数据可视化与科学研究

尽管 HTML5 Canvas 游戏和 WebGL Demo 在互联网上吸引了绝大多数人的眼球,但在这之外,依然有人默默地致力于研究如何将 HTML5 WebGL 利用于医学和科研领域的的数据可视化。

例如 XTK,全称是 X Toolkit,就是一个被设计为用于科研领域数据可视化的 WebGL 框架。它可以解析标准的科研数据格式,并生成全交互的、可以旋转的、可以自由查看的 3D 模型。

它可以被广泛利用于各个领域,从心脏扫描的 3D 图形演示,到 CERN 大型粒子对撞机的散布形态,都可以被很好地进行可视化输出。在图形输出上,XTK 主要致力于体绘制技术,它允许进行实体建模(CSG: Constructive Solid Geometry);也就是说,基本上可以用最简单的几何体形状根据集合论的布尔逻辑来组成复杂物体。在处理 3D 图形数据的横断面切片时,非常简单易用。



XTK 的作者 Daniel Haehn 和 Nicolas Rannou 之前从事于传统桌面领域的的数据可视化工程,在去年夏天偶然看到了包括 WebGL Brain Viewer 在内的一些 WebGL Demo 之后,他们开始研究那些基于网络的数据可视化工具。

“那些工具通常都要求使用者对低级计算机图形编程有深厚的了解,这对于(科研领域的)目

标人群来说学习难度太大了!”

于是,仿照基于 C++ 开发的数据可视化框架,XTK 就诞生了。它的目标是尝试创建一套可以应用于多个领域并且简化处理流程的 API。关于 XTK 的更多信息,你可以查看 Daniel Haehn 的题为《WebGL for Baby Brains》的精彩演讲,其中讲述了如何在神经成像领域更好地利用 web 技术。

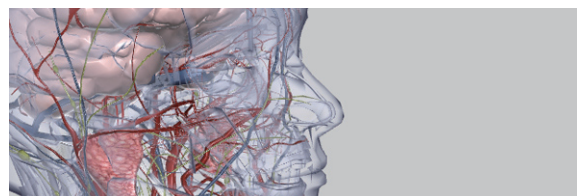
其他科研可视化项目 Zygote Body



这个之前被大家熟知为“Google Body”的项目大概是最为著名的使用了 WebGL 技术的医学可视化范例。

它以完全互动的方式将完整的人体结构分层呈现在人们面前,包括肌肉结构、各个器官和心血管系统。现在 Google 已经将此项目过继给 Zygote 公司,而且公开了除美术资源之外的所有源代码。

BioDigital Human

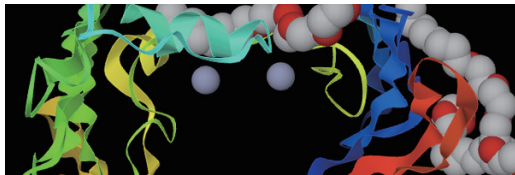


BioDigital Human 使用了 SceneJS 这个 WebGL 框架创建了更具细节的人体解剖演示。目前该项目刚刚结束 beta 测试,仍然有一些 bug,

HTML 5 数据可视化与科学研究 II

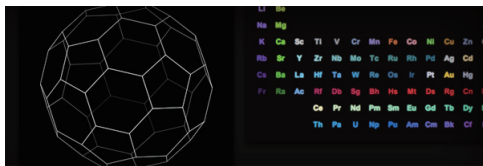
但是当你可以正常的运行时,你一定会为它感到震惊不已。

WebGL Molecule Viewer



科研领域的数据可视化并不只是人体结构,例如 WebGL Molecule Viewer 就可以解析符合工业标准的分子描述格式,然后输出一个可以交互的 3D 模型查看界面。这个项目的网站看起来有些简陋,但是内在功能却十分强大,值得一看。

CanvasMol



最后要介绍的是 AlteredQualia 在去年夏天发布的作品, AlteredQualia 同时也是著名的 WebGL 框架 Three.js 的主要代码贡献者之一。CanvasMol 可以和上面介绍的 WebGL Molecule Viewer 一样读取工业标准格式,然后生成可交互的 3D 模型,但是它的界面做的更加美观。

未来将 HTML5 WebGL 技术应用于文化教育领域的潜力十分巨大。与奢侈的游戏娱乐行业相比,这些团体和组织通常资金有限,但却拥有海量的科研数据。如果能够通过各种工具让这些数据内容显示在浏览器中,那么它们将再次焕发新生。想象一下——在一个博物馆的网站上,访问者可以在浏览器中自由的查看每一件展品;或者在某个医学院的图书馆中而不是实验室里,学生可以在“数码尸体”上下手,学习解剖学知识。

51CTO 精品杂志推荐《Linux 运维趋势》

《Linux 运维趋势》是由 51CTO 系统频道策划、针对 Linux/Unix 系统运维人员的一份电子杂志,内容从基础的技巧心得、实际操作案例到中、高端的运维技术趋势与理念等均有覆盖。



本期为 2012 年 3 月号,总第 18 期,主题为运维安全。

本杂志长期处于探索期,需要更多来自大家的意见与参与。如果您对这份电子杂志感兴趣,那就请您下载阅读;想要帮助我们做的更好,请进入我们的 Linux 运维趋势技术圈留下您的宝贵意见建议。

读者讨论组: <http://g.51cto.com/linuxops/>

邮件订阅入口:

<http://os.51cto.com/art/201011/233915.htm>

投稿邮箱: yangsai#51cto.com

发布周期: 每个月的第二个星期五

往期《Linux 运维趋势》下载汇总页:

<http://down.51cto.com/zt/71>

先为最近几期的跳票向大家道声抱歉。最近由于人手不足,杂志这边的工作常有耽误。如有对 InDesign 或电子出版比较了解的朋友愿意帮忙的,欢迎联系 yangsai#51cto.com, 谢谢!

下载地址(51CTO 下载频道):

<http://down.51cto.com/data/329527>

■ 编者按

这次来讲 MVC 中最后的 M。Model, 几乎可以说是网页应用的核心。之前课程提到过网页应用是由数据库驱动, 而在很多场景, 数据库 = M ; M = 数据库。

宅男程序员给老婆课程9：数据模型

所谓的 ORM; object relational mapping。

现在新的网页开发框架, 特别是 MVC 框架, 都会提供 ORM 支持, 避免程序员直接写 SQL、操作数据库。

传统上, ASP/ php 臭名昭著的 sql 注入问题, 便是因为菜鸟程序员直接在程序中根据用户输入拼接数据库造成的; 而使用 ORM 框架, 则可以彻底避免这种问题。

ORM 有两种风格, 一种是 $R \Rightarrow O$; 一种是 $O \Rightarrow R$ 。

===== $R \Rightarrow O$ =====

传统上, 程序员也都是先完成数据库设计 (甚至是由 DBA 完成), 然后再考虑相应的对象生成, 也就是所谓的 $R \Rightarrow O$ 。

在这样的场景下, 整个软件的框架, 还是以数据库为核心, 业务的设计思维是以关系型数据库的表结构为基础去考虑的, 具体应用实现上, 会考虑很多关系型数据库的功能特性, 比方说, 外键, joining 等等, 并且, 程序员需要直接考虑“数据库设计三范式”, 以及冗余字段等面向数据库的优化手段。

并且, 程序员也很可能会采用数据库的一些高级特性, 如视图、存储过程、甚至触发器等等以方便使用。

O 的存在, 仅仅是为了方便操作数据库表。

===== $O \Rightarrow R$ =====

这种设计哲学则是相反, 程序员做业务分析、实现架构设计的时候, 并不过多考虑数据库的特性与限制; 程序仅考虑自己的业务对象: 编程语言中的对象。

数据库仅仅只是作为一个对象持久层来考虑:

- ★ 程序运行的时候, 对象是自动保持在内存中。

- ★ 但在对象状态改变、程序退出的时候, 将对象保存进数据库。

- ★ 程序重新启动的时候, 则从数据库中获得原先数据, 并还原为内存中的对象。

在这样的场景下, 数据库是一个可以被替换的存储层, 它可以是关系型数据库, 也可以是 NoSQL, 甚至是硬盘文件; 所以, 即便使用关系型数据库, 一般也不会使用其高级功能。

设计哲学的不同直接造成了使用技术的不同。

===== 比较 =====

在 ED 开发圣经 PEAA 中, 列举了下面三种方案:

1. Trascation Script

也就是直接拼接 SQL 啦~

2. Table Module

$R \Rightarrow O$; 并且, O 非常简单, 直接以类似数组的方式读取表数据。

宅男程序员给老婆课程 9: 数据模型 II

.net 中 ADO.net 的 DataTable / DataRow 对象便是这种设计的典型实现。

3. Domain Model

O => R, 直接设计业务领域 (Domain) 的对象, 然后在考虑对象的持久化方案。

针对上面 3 中方案, Martin Fowler 画了下面这张著名图解:

<http://www.hamishgraham.net/page/Work-Habits.aspx/Architecture/Business-Layer>

他的结论是:

使用 Table Module 的方式, 永远比直接写 SQL 简单; 在简单的业务场景下, Table Module 也会比 Domain Model 简单, 但 Table Module 的方案复杂度会随着业务复杂化而快速增长。

反之, Domain Model 的复杂度跟业务复杂度相比始终保持水平增长; 它虽然一开始最复杂, 但随着业务复杂度超过一定程度后, 它反而会成为最简单的方案。

就我自己的开发经验, 基本与 Fowler 的描述吻合; 但随着 ORM 技术的成熟, Domain Model, 未必如他在图中画的那样, 一开始就有那么高的复杂度。

关键是看程序员是否习惯于关系型数据库的实现方案, 如果是, 那么, 切换去 Domain Model, 确实会比较麻烦, 各种不适应。

但如果是一个没有关系型数据库经验的程序员, 或者说, 没有强制使用 SQL 思维习惯的程序员, 使用 Domain Model, 也可以是很自然的方案。

下一课, 讲继续详细说明 Domain Model。

作业:

1. ED 开发圣经 PEAA 究竟是哪本书?

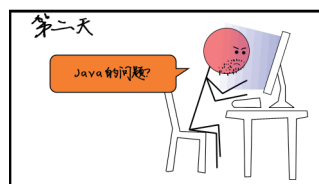
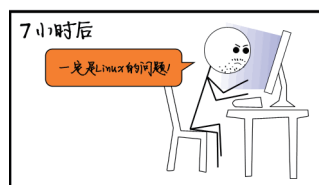
2. 数据库三范式是什么?

3. 关于 Domain Model, 什么是充血模型? 什么是贫血模型? ■

男主角: Wuvist (新浪微博), 真名翁伟, 自称胖程序员一个, 幸好已婚。学习 .net 出身, 现常用 python 做服务器端开发, 曾任新加坡某创业公司主程。公司被 techcrunch blog 过后, 觉得新加坡生活太过安逸, 终于于去年辞职只身回家乡汕头创业, 活跃于珠三角技术沙龙, 热衷于与其他技术宅分享。

女主角: Katze, Wuvist 的老婆, 女程序员, 在某跨国投行任 Unix 系统管理员, 常被 Wuvist 嘲笑技术太差。

总之, 因 Wuvist 只身回国创业, 这对分隔天涯的技术宅男宅女竟然想出了定期写技术课程、交作业这种方式来保持联系, 这何止是令人发指? 简直就是令人发指! ■



为什么我不再做.NET开发

写这篇博客已经等待很长时间了。我最初开始做网站开发是使用 ASP .NET 和 C# 技术。我很荣幸分别在 2010 和 2011 年被微软授予了“全球最有价值专家”（即 MVP）称号。我想在等待荣誉过期到来之前离开 .NET 开发的世界。

我的职业生涯是伴随着 .NET 社区开始,我的文章近来很少被微软推荐。我可以想象到,相对于其他更让人好奇的 C# 文章,大家对曾经喜欢过我的 “Web Forms to MVC” 热情已经消失。我没有失去什么,我会永远的感激 .NET 社区,因为他们帮助我得到了现在拥有的东西。有几个原因导致我不在做专业的 .NET 开发。我将会在这篇博客中尝试去解释一下。

首先是第一件事。

我不得不退出我很有经验的 .NET 开发,虽然这些是非常美妙的事情。.NET 社区非常的热情并且开放。.NET 开发是我见过的男女比例最高的开发环境。这一现象的原因是显而易见的:人。当我是新人时,他们都非常欢迎我,他们帮助我学习很多东西。我真不知道如何感谢他们。

这里入门的门槛很低,这里很少有“其实...”,更多的是“欢迎!”我被我的朋友们的积极鼓励作出贡献。当我回到那里遇到很多人的时候,仍然称他们为朋友。我希望像一些前辈那样在将来带着更多人再回到那里。

愤怒的咆哮

我看到过很多人“退出”.NET 开发世界。通常经常会引起一个很长的 Twitter 的战争,愤怒的留言,微软不支持他们的开发人员,导致他们的团

队失败。工具很难用,语言的说明很荒唐,支持不够迅速,某人是个混蛋,等等。它总是让我回到留言板,很多人因此退出了网络。

人们退出的主要因素可能因为一些特别的原因,或者指南不明确,或者其他这种表面的原因,这是对微软做为一个商业公司的错误的理解。微软是为了赚钱的公司。微软要为股东的利益负责。因此,C#、F#、ASP.NET 以及所有其他微软的框架/语言都是为了创造利润而经营。是的,我知道他们主要的收入并不是来自于这些,甚至不是来自于 10,000 美元的 MS SQL 的服务授权许可。无论如何,他们所有的雇员,部门,以及其他所有的公司人员都要给他们董事会一个交代。

开发者们不能理性的对待,他们不会带领大家去做更完美的工具或者一个更好的社区。做为一个公司花钱为大家提供了大量资源,但是,最终,你的动力是在你所创造的东西中。

什么是我必须做的

好吧,做为一个开发者,微软在支持我。为我去拜访他们付费,甚至我得到了很漂亮的夹克。为什么要破坏这些东西呢?对我来说,这些是做人的一些基本准则。■

更多内容请参考原文:

<http://developer.51cto.com/art/201204/327003.htm>

传统程序员的出路在何方？

今年的百度开发者大会实在可以用人山人海来形容。在庞大的会场中挤满了年轻的人们，他们大多是快速的 Web 开发者和移动 App 开发者，怀抱着改变世界的热情，希望在这次云计算和移动互联网的大潮中一展身手。对于那些传统行业的程序员们，这次大潮能为他们带来些什么？他们又将如何投入到这次大潮当中？

“PC 的互联网时代是站长的时代，那时的程序员们在做企业的信息化，大多数的程序员没能赶上 PC 的时代。”

“而移动互联网，将是属于开发者的时代。”

在 3 月 23 日的百度开发者大会上，李彦宏在开场致辞中这样说到。

今年的百度开发者大会实在可以用人山人海来形容。在庞大的会场中挤满了年轻的人们，他们大多是快速的 Web 开发者和移动 App 开发者，怀抱着改变世界的热情，希望在这次云计算和移动互联网的大潮中一展身手。

然而，软件行业、甚至互联网行业本身已经发展了很长一段时间，作为一个趋于成熟的行业，这个世界并非只有年轻人。如果那些 90 年代、甚至 80 年代开始进入软件行业的老人们普遍感到难以立足、自己的价值得不到体现，那么整个行业的生命力实在令人堪忧——这说明它没有办法持续的带领这批人去成长，整个行业只是一个没有沉淀的浮萍，风一吹云就散了。

“现在经常会感到很吃力。我是做工控软件的，一直在微软平台上做 C 语言开发，像是这些网站、移动应用的开发，虽然正在学着做，但我在这个领域很难建立什么技术优势啊。”

笔者在会场遇到一位李先生，他在聊天的时

候这样说到。

“你看，其实在工控领域这么多年，客户最根本的需求一直没什么变化，就是通过设备上的传感器监控设备，设备有状况时能够及时给出提示或警告，能够手动或自动对设备进行启动、关闭等操作等等。我一直在用 C 语言开发，用户操作界面一直用 ActiveX 的方式来实现，功能上的需求都可以满足。”

“那您这次来百度开发者大会，主要关注什么呢？”台上的林仕鼎开始介绍百度的开发者平台，看起来都跟工控软件的世界没什么关系。

林仕鼎“我这也算是被逼的没办法了吧？不学习新东西不行啊！”李先生笑了笑。“其实以前我们也遇到过问题，就是有些客户要求我们的软件能运行在 Unix 之上。这种需求我们能理解，因为 Unix 在这个领域也算是身份的象征嘛，你的企业用 Unix，跟别人说起来比较有面子。但是我们的软件用很多微软平台的类库开发，做个 Unix 版本没那么容易。”

电话声响起，李先生掏出一台中兴智能手机应答了几句，回过头来继续刚才的话题。

“以后也许客户还会要求在其他平台上能运行的系统，比如智能手机和平板，所以需要赶紧了解这方面的知识。另外就是使用情景的变化。”

传统程序员的出路在何方？ II

李先生继续说到。“以前我们用 ActiveX,这个东西既不能跨平台也不安全,不过在企业内部使用倒是问题不大,毕竟这个行业基本没有什么在工厂外面做远程管理的需求。

但未来的事情有很多变数,有些企业可能会想要在外面也能监控设备,或者因为看到某些很炫的功能而提出需求,那么单凭以前的技术积累肯定是没有竞争力的。所以我现在也在学习 C# 语言,虽然目前工作上没太用到,但也能做出一些东西了。”

“为什么不去学一些更加跨平台的语言呢?”想到对方已经因为微软的专有类库吃过软件无法移植的亏,却仍然选择了另一个微软的专有技术,笔者感到有些诧异。

“Web 上常用的 PHP、Python、Ruby 都是很快捷的语言,也可以处理复杂的需求,当然也可以用 Java。前端的话现在有 HTML 5,这个技术目前还不够成熟,生产上的项目最好保持一定距离,但很建议个人去玩玩。iOS 的话目前都是用 Objective-C 来开发,但是用 HTML 5 也行,包括现在微软的 Windows 8 也开始接受 HTML 5 的应用呢。”

“可能我还是觉得这方面不是我的竞争力所在吧?”台上的郑博闻开始分享了,那是一位年仅 13 岁的移动 App 开发者。

郑博闻“我做这么多年,积累的经验都在工控软件领域。如果跳出这个领域跟这些年轻人拼,确实没什么优势。我对 Web 开发的这些东西了解的太少了。”

不过,李先生明显对 HTML 5 这些新技术产生了兴趣,打听了一些这方面的技术细节。

“有没有考虑在业余时间做点东西呢? 做点给消费者使用的软件。”

“有啊,我平时也一直在观察这个移动 App 的市场。感觉要做的话,母婴这个领域的发展空间会不错。”李先生提到自己有一个还没到上小学年龄的儿子。“做父母的,总是愿意在孩子身上花钱,而且花的很开心。现在国内做应用赚钱还是很艰难的,但是母婴这一块,至少消费者的心理壁垒会比较弱一些。”

移动互联网的时代真的来了,发展的势头已经让业内的每一个人都无法忽视它。如果说当年 PC 的崛起是创造了一个新的市场,那么移动互联网的崛起,则随着其上亿级、上十亿级的用户数量,完全改变了很多传统行业的运作方式。

它正有意无意的迫使所有的人去适应它的存在,其门槛之低,无论是还在上中学、小学的孩子,还是之前没有摸过代码的家庭主妇,都可以进来一拼。

对于经验丰富的传统程序员而言,如果不是本身对软件行业没兴趣,是绝对不会忽视这次大潮的。

传统行业的需求注定将要改变,现在意识到这一点还不晚。对行业的了解是传统程序员们的优势之一,如果能够用移动开发和云计算开发的技能武装自己,学习一些快捷灵活的语言,则能够得心应手的应对即将到来的变化。■

作者 / 杨赛

■ 编者按

MySQL Proxy 是一个处于 MySQL Client 端和 MySQL Server 端之间的简单程序,它可以监测、分析或改变它们的通信。它使用灵活,没有限制,常见的用途包括:负载均衡,故障、查询分析,查询过滤和修改等等。

Centos5 下 MySQL Proxy 0.8.2 安装

MySQL Proxy 就是这么一个中间层代理,简单的说,MySQL Proxy 就是一个连接池,负责将前台应用的连接请求转发给后台的数据库,并且通过使用 lua 脚本,可以实现复杂的连接控制和过滤,从而实现读写分离和负载均衡。对于应用来说,MySQL Proxy 是完全透明的,应用则只需要连接到 MySQL Proxy 的监听端口即可。当然,这样 proxy 机器可能成为单点失效,但完全可以使用多个 proxy 机器做为冗余,在应用服务器的连接池配置中配置到多个 proxy 的连接参数即可。

MySQL Proxy 更强大的一项功能是实现“读写分离”,基本原理是让主数据库处理事务性查询,让从库处理 SELECT 查询。数据库复制被用来把事务性查询导致的变更同步到集群中的从库。在生成 Mysql 的 M-S 结构后,为了读写分离,需要使用 Mysql Proxy。目前 Mysql Proxy 的版本为 0.8.2。

MySQL Proxy 目前应用的生产环境:

一、现实中很多应用环境是这样:在程序开发初期就没有规划好,程序读写数据都只提供一个 IP 接口,后期由于压力顶不住,公司要求后来的系统架构师 /DBA 将公司的数据库架构更改,这样在程序上实现比较困难,所以加进此中间层来实现。

二、建议大家还是将其应用中小型网站,特别是读比较频繁的网站。如果是大型网站或门户型的网站,建议读写分离还是从开发层面实现。

三、建议大家采用 MySQL Proxy 0.8.0 以上的版本,它修正了以前不少 BUG,而且稳定性也不错。

一、MySQL Proxy 0.8.2 的安装步骤如下:

先从 mysql.com 官方地址 <http://dev.mysql.com/downloads/mysql-proxy/> 下载源码包到 /usr/local/src/ 目录下,建议大家采用最新的 MySQL Proxy 0.8.2 版本,MySQL Proxy 0.8.2 安装之前有先决条件,如下:

libevent 1.x 或更高

glib2 2.6.0 或更高

lua 5.1.x 或更高

pkg-config

libtool 1.5 或更高

MySQL 5.0.x 或更高的开发库

服务器操作系统: Centos 5.6 x86_64

1. 为了加速安装进度,我们可以先 yum 安装必须的库,同时解决 pkg-config、libtool 和 Mysql 开发库,由于 mysql-proxy 实际并不需要在本机上运行 mysql 实例,所以我们在这里用 yum 安装,命令如下所示:

```
yum -y install gcc gcc-c++ autoconf mysql-devel libtool  
pkgconfig ncurses ncurses-devel
```

2. libevent 安装 libevent-2.0.13 版本,从此处可以下载:

<http://monkey.org/~provos/libevent-2.0.13-stable.tar.gz>

运行如下命令进行安装:

```
tar xvf libevent-2.0.13-stable.tar.gz cd  
libevent-2.0.13-stable ./configure make && make  
install
```

Centos5.6 下 MySQL Proxy0.8.2 的安装及测试 II

3.glib2 安装 glib-2.18.4 版,从此处可以下载:

<http://ftp.gnome.org/pub/gnome/sources/glib/2.18/glib-2.18.4.tar.gz>

运行如下命令进行安装:

```
tar xvf glib-2.18.4.tar.gz cd glib-2.18.4 ./configure make && make install
```

4.lua 安装 5.1.4 版本,安装之前需要先安装 readline 6.1,不然会报错缺少头文件:

readline 6.1 下载:

<ftp://ftp.cwru.edu/pub/bash/readline-6.1.tar.gz>

运行如下命令进行安装:

```
tar xvf readline-6.1.tar.gz cd readline-6.1 ./configure make && make install
```

为了让动态链接库为系统所共享,我们这里用 ldconfig

ldconfig -v 用此选项时,ldconfig 将显示正在扫描的目录及搜索到的动态链接库,还有它所创建的连接的名字。

5. 安装 lua 5.1.4,其下载地址为 <http://www.lua.org/ftp/lua-5.1.4.tar.gz>。

```
cd /usr/local/src tar xvf lua-5.1.4.tar.gz cd lua-5.1.4
```

64 位系统,需在 CFLAGS 里加上 -fPIC,我们用 vim 编辑下 src/Makefile 文件,修改代码如下所示:

```
CFLAGS= -O2 -Wall -fPIC $(MYCFLAGS)
```

我们继续用如下命令进行安装:

```
make linux make install
```

6. 配置 pkg-config 环境变量,命令如下所示:

```
cp etc/lua.pc /usr/local/lib/pkgconfig/ export PKG_CONFIG_PATH=$PKG_CONFIG_PATH:
```

/usr/local/lib/pkgconfig 7. 以上操作就完成了基础文件的安装,接下来是 MySQL Proxy0.8.2 安装,下载完后运行如下命令:

```
tar xvf mysql-proxy-0.8.2.tar.gz cd mysql-proxy-0.8.2 ./configure prefix=/usr/local/mysql-proxy make && make install cp lib/rw-splitting.lua /usr/local/lib/ cp lib/admin.lua /usr/local/lib/
```

二、到这里 MySQL-proxy 已基本安装完成,接下来就是 MySQL Proxy0.8.2 的测试和配置了。

MySQL Proxy 环境设置说明

Master MySQL 服务器: 192.168.2.117

Slave MySQL 服务器: 192.168.192.168.2.115

MySQL Proxy 服务器: 192.168.2.112

1.mysql-proxy 选项说明,大家也可以用 mysql-proxy help-all 查看它的帮助选项,命令如下所示:

mysql-proxy --help-all 管理功能选项:

--admin-address=host:port 指定一个 mysql-proxy 的管理端口,缺省是 4041;
--admin-username=<string> username to allow to log in
--admin-password=<string> password to allow to log in
--admin-lua-script=<filename> script to execute by the admin plugin 代理功能选项:

-P, --proxy-address=<host:port> 是 mysql-proxy 服务器端的监听端口,缺省是 4040,建议改为 3306,方便开发人员写代码。

-r, --proxy-read-only-backend-addresses=<host:port> 只读 Slave 的地址和端口,

Centos5.6 下 MySQL Proxy0.8.2 的安装及测试 III

缺省为不设置;

`-b, --proxy-backend-addresses=<host:port>`

远程 Master 地址和端口,可设置多个做 failover 和 load balance,缺省是 127.0.0.1:3306;

`--proxy-skip-profiling` 关闭查询分析功能,缺省是打开的;

`--proxy-fix-bug-25371` 修正 mysql 的 libmysql 版本大于 5.1.12 的一个 #25371 号 bug;

`-s, --proxy-lua-script=<file>` 指定一个 Lua 脚本来控制 mysql-proxy 的运行和设置,这个脚本在每次新建连接和脚本发生修改的时候将重新调用;

其他选项:

`--defaults-file=<file>` 配置文件,可以把 mysql-proxy 的参数信息置入一个配置文件里,建议大家用这种配置 MySQL Proxy0.8.2,比较方便;

`--daemon mysql-proxy` 以守护进程方式运行;

`--pid-file=file` 设置 mysql-proxy 的存储 PID 文件的路径;

`--keepalive try to restart the proxy if it crashed`,保持连接启动进程会有 2 个,一号进程用来监视二号进程,如果二号进程死掉自动重启 proxy,这是新版 MySQL Proxy 的增加的 Keepalived 功能,它修正了以前 MySQL Proxy 容易死掉的 bug,建议大家开启此功能。

完整的 mysql-proxy 配置文件如下:

<http://database.51cto.com/art/201203/324475.htm>

给用户授权

在 Master/Slave 建立一个测试用户,因为以

后客户端发送的 SQL 都是通过 mysql-proxy 服务器来转发,所以要确保可以从 mysql-proxy 服务器上登录 MySQL 主从库,分别在主和从 MySQL 机器上执行如下所示:

```
mysql> grant all privileges on *.* to 'test'@'192.168.2.117' identified by 'test' with grant option;
mysql> grant all privileges on *.* to 'test'@'192.168.2.115' identified by 'test' with grant option;
```

3. 做完此步我们就可以在 MySQL Proxy0.8.2 上进行测试了,命令如下所示:

```
/usr/local/mysql-proxy/bin/mysql-proxy -P 192.168.2.112:3306 --defaults-file=/etc/mysql-proxy.cnf
```

我们监测下 MySQL Proxy 的日志,有如下字样表示 MySQL Proxy 已成功启动了(注意之前要成功启动 117 和 115 上的 mysql 程序),如下所示:

<http://database.51cto.com/art/201203/324475.htm>

我们多开些 MySQL 客户端,会发现 MySQL Proxy0.8.2 很容易的利用 Lua 脚本实现读写分离功能了,在进行测试前我比较担心的一个问题是:如果主 Master 因故障停止服务了,MySQL Proxy 会不会在从机上面写数据呢?这样会导致主从数据不一,事实上,我在停掉 192.168.2.117 上的 MySQL 时发现,MySQL Proxy0.8.2 直接不允许写数据了。

三、MySQL Proxy0.8.2 测试中遇到的问题



本文未完,详细部分请参考原文

<http://database.51cto.com/art/201203/324475.htm>

IBM 副总裁 Judy: 下代 DB2 年内发布

2012 年 3 月 16 日, IBM DB2 数据库迁移之星大赛在盘古大观酒店举行。51CTO 记者有幸在大会上采访了 IBM 副总裁 Judy Huber 女士,就大数据处理和下一代 DB2 产品进行了探讨。

2012 年 3 月 16 日,2012 IBM DB2 数据库迁移之星大赛在盘古大观酒店举行。在此次比赛中共决出了七个优秀奖和冠亚季军,获得前三名的对于将获得前往美国参观学习的机会。

就如同 iPad 3 是不是应该叫 iPad new 一样, Judy 女士首先表示 IBM 团队正在为新一代 DB2 数据库日夜努力工作,但下一代 DB2 产品叫不叫 DB2 10 还没有确定。但是这个版本的 DB2 数据库将会带来全新的变化,而不仅仅是 9.7 升级到 9.9 这样的程度。同时可以肯定的是,新产品将会在 2012 年年内推出。



IBM 副总裁 Judy Huber 女士(51CTO 摄)

在此次 DB2 迁移之星大赛上,我们能感受到一些银行机构也在考虑尝试从 Oracle 迁移到 DB2 数据库。但作为银行方面,该如何考量迁移过程中的风险? 是否值得将自己的核心业务数据库进行迁移? IBM 能做出那些举措让客户成功

避免这些风险?

Judy 女士告诉 51CTO 记者,作为大型金融机构不会考虑为了迁移数据库而迁移数据库,它一定是因为有了更大的需求而需要去移植。无论是何种系统的迁移,特别是业务主系统的迁移都是带有一定风险。所以 IBM 通常都会帮助自己的客户进行非常仔细具体的规划,同时在实施的过程中提供一些技术和工具以帮助客户规避风险。对此,IBM 中国开发中心信息管理软件总经理朱辉先生还进行了补充,他表示客户一般都不会一上来就要求迁移自己的主营业务系统,做一个这么大的迁移决定。尤其是那些对于 IBM 的技术和产品效能都不是非常熟悉的客户,不会大动干戈。但是往往他们会通过一些不是那么重要或者跟自己的技术团队比较长时间的使用后,对于 IBM 的技术不是那么恍惚的时候就会有更深层次的合作了。

数据库安全问题一直是广大 DBA 比较关心的问题,特别是去年年底的中国互联网网站大规模泄密事件给广大用户造成了不可避免的伤害。51CTO 记者想从 Judy 女士方面知道下一代 DB2 数据库产品在安全性方面会不会有更多的保障。

本文未完,更多内容请参考原文 ■

<http://database.51cto.com/art/201203/323813.htm>

从个人网站到淘宝网 仰观Java时代淘宝的技术发展

从 2003 年的一个个人对个人(C2C)的商品交易网站到如今的淘宝网,其实在作为个人网站发展的时间里并不长。那么在这段时间里,淘宝究竟是如何发展的呢?在这篇文章里我们将找到淘宝网的发展历史以及所用到的技术。

引言

光棍节的狂欢

“时间到,开抢!”坐在电脑前早已等待多时的小美一看时间已到 2011 年 11 月 11 日零时,便迫不及待地投身于淘宝商城一年一度的大型网购促销活动——“淘宝双 11 购物狂欢节”。小美打开早已收藏好的宝贝——某品牌的雪地靴,飞快的点击购买,付款,一回头发现 3000 双靴子已被抢购一空。

小美跳起来,大叫一声“欧耶!”

小美不知道,就在 11 日零点过后的这一分钟内,全国有 342 万人和她一起涌入淘宝商城。当然,她更不知道,此时此刻,在淘宝杭州的一间办公室里,灯火通明,这里是“战时指挥部”,淘宝技术部的一群工程师,正在紧盯着网站的流量和交易数据。白板上是他们刚刚下的注,赌谁能最准确地猜中流量峰值和全天的交易总额。他们的手边放着充足的食物和各类提神的饮料。

一阵急促的电话声响起来,是前线部门询问数据的,工程师大声报着:“第 1 分钟,进入淘宝商城的会员有 342 万”。过一会工程师主动拿起电话:“交易额超过 1 亿了,现在是第 8 分钟。”接下来,“第 21 分钟,刚突破 2 亿”。“第 32 分钟,3 亿了”。“第 1 个小时,4.39 亿”。这些数据随后出

现在微博上,引起一片惊呼。

“完蛋了!”突然有人大喝一声,所有的眼睛都紧张的盯着他,只见他挠挠头,嘿嘿的笑道“我赌的少了,20 亿轻松就能过了,我再加 5 亿”,他跑去白板边上把自己的赌注擦去,写上 25,接下来有人写上 28,有人写上 30,有人跑到微博上开下盘口,同事们纷纷转载下注。接下来的这 24 个小时,战时指挥部的工程师们都不能休息,他们盯着网站的各种监控指标,适时的调整机器和增减功能。顶住第一波高峰之后,这些人开始忙里偷闲的给自己买东西,大家互相交流着哪家买的移动硬盘靠谱,哪家衣服适合自己的女朋友,不时的有人哀嚎宝贝被人抢了、信用卡额度不够了。同时,旁边白板上的赌注越下越大。



从个人网站到淘宝网 仰观 Java 时代淘宝的技术发展 II

11月11日,这个棍子最多的日子被网民自我调侃的变成了一个节日——“光棍节”。而淘宝网又用疯狂的折扣促销给它赋予了另外一个意义——“购物狂欢节”。2011年11月11日这一天,淘宝商城与淘宝网交易额之和突破52亿,这个数字是“购物天堂”香港一天零售总额8.5亿的6倍。

网民感受到的是疯抢的喜悦,而网站的技术人员感受到的却是“压力山大”。就如同你家办酒席,宴请左邻右舍,这个办起来容易。倘若宴请十里八乡所有的人,吃饭的人自然开心,但却不是一般人家能够办得起来的。能办得起来如此盛宴者,需要强大的财力物力、组织能力、技术实力(例如做这么多菜,你的炒锅一定要是“分布式的”、“可复制的”、“可扩展的”,洗菜切菜要有“工作流引擎”,上菜的路径要用图论来计算出来,甚至连厨房的下水道都要重新设计)。

淘宝能够举办如此盛宴,网站的技术实力可见一斑。淘宝网拥有全国最大的hadoop分布式计算集群之一,日新增数据50TB,有40PB海量数据存储。分布在全国各地80多个节点的CDN网络,支持的流量超过800Gbps。淘宝的搜索引擎能够对数十亿的商品数据进行实时搜索,另外还拥有自主研发的文件存储系统和缓存系统,以及java中间件和消息中间件系统,这一切组成了一个庞大的电子商务操作系统。另外从商业数据上来看,AMAZON的财报显示2011年完成了大约480亿美金的交易额,EBAY2011年财报全年完成了大约600亿美金的交易额(不包括其独立的汽车交易平台)。不管从交易额、商品数量、同比增速等指标上看,淘宝网均远超于此,是目前全

球最大的电子商务平台。(由于淘宝非上市公司,未公布2011年业绩,以上内容来自淘宝网技术副总裁@_行癫的微博)

以上这些技术数据可能已经让一些同学产生不适的感觉,为了让更多的人读懂这本书,我们从技术的角度来看,小美访问淘宝网的时候,网站上发生了什么事情。下参考资料:《你刚才在淘宝上买了一件东西【技术普及帖】》,来自南京邮电大学孙放同学。

为了有个更直观的对比,我们说一个同行,他在2011年光棍节之前做促销,流量上去之后,达到12Gbps(他们有这么大的流量,老板很高兴,在微博上面说了这个数据),这时候流量达到了极限,网站几乎挂掉,用户无法下订单。而淘宝网光棍节当天网络的流量最高达到800多Gbps,带给各家银行和快递公司的流量也让他们压力山大,如临大敌(后来,他们以能够撑住淘宝带来的流量为荣而到处宣传)。另外如果你在网上购买过火车票的话,更能体会到网站能支持多大的流量有多重要。但这不是一朝一夕做出来的,也不是有钱就能办到的。

以上对比的这些网站,也许读者很容易就猜到是哪一家,这里拿出来作对比,绝对没有嘲笑人家的意思,采用通常的网站技术方案,能做到这种程度已经不错了。任何网站的发展都不是一蹴而就的,在什么样的阶段采用什么样的技术。在发展的过程中网站会遇到各种各样的问题和业务带来的压力,正是这些原因才推动着技术的进步和发展,而技术的发展又会反过来促进业务的更大提升。二者互为因果,相互促进。如今淘宝网的

从个人网站到淘宝网 仰观 Java 时代淘宝的技术发展 III

流量已经是全球排名第 12、国内排名第 3(美国的 ebay 全球排名 23,国内前两名是百度和腾讯)。淘宝网的系统也从使用一台服务器,到采用万台以上的服务器。本书就为大家描述淘宝网在整个发展过程中,所有的主动和被动的技术变革的前因后果,这由很多有趣的故事组成。

正如同很多人或组织成功了以后,就会为自己的出身编造一个美丽的传说。淘宝网的出身,网上也有非常多的传说,下面我们就从它的出生开始讲起。

个人网站

2003 年 4 月 7 日,马云,在杭州,成立了一个神秘的组织。他叫来十位员工,要他们签了一份协议,这份协议要求他们立刻离开阿里巴巴,去做一个神秘的项目。这个项目要求绝对保密,老马戏称“连说梦话被老婆听到都不行,谁要是透漏出去,我将追杀到天涯海角”。这份协议是英文版的,匆忙之间,大多数人根本来不及看懂,但出于对老马的信任,都卷起铺盖离开了阿里巴巴。

他们去了一个神秘的据点——湖畔花园小区的一套未装修的房子里,房子的主人是马云。这伙人刚进去的时候,马云给他们布置了一个任务,就是在最短的时间内做出一个个人对个人(C2C)的商品交易的网站。现在出一个问题考考读者,看你适不适合做淘宝的创业团队。亲,要是让你来做,你怎么做?

在说出这个答案之前,容我先卖个关子,介绍一下这个创业团队的成员:三个开发工程师(虚竹、三丰、多隆)、一个 UED(二当家)、三个运营(小宝、阿珂、破天)、一个经理(财神)、还有就是

马云和他的秘书。当时对整个项目组来说压力最大的就是时间,怎么在最短的时间内把一个从来没有的网站从零开始建立起来?了解淘宝历史的人知道淘宝是在 2003 年 5 月 10 日上线的,这之间只有一个月。要是你在这个团队里,你怎么做?我们的答案就是:买一个来。

买一个网站显然比做一个网站要省事一些,但是他们的梦想可不是做一个小网站而已,要做大,就不是随便买个就行的,要有比较低的维护成本,要能够方便的扩展和二次开发。那接下来就是第二个问题:买一个什么样的网站?答案是:轻量一点的,简单一点的,于是买了这样一个架构的网站:LAMP(linux+apache+mySQL+PHP)。这个直到现在还是一个很常用的网站架构模型。这种架构的优点是:无需编译,发布快速,PHP 功能强大,能做从页面渲染到数据访问所有的事情,而且用到的技术都是开源的,免费。

当时我们是从一个美国人那里买来的一个网站系统,这个系统的名字叫做 PHPAuction(他们的官方网站 <http://www.phpauction.net> 这个名字很直白,一眼就看出来这个系统是用什么语言做的、是干什么用的),PHPAuction 有好几个版本,我们买的是最高版的,功能比较多,而且最重要的是对方提供了源代码。最高版比较贵,花了我们 2000 美金(貌似现在降价了,只要 946 美元)。买来之后不是直接就能用的,需要很多本地化的修改,例如页面模板改的漂亮一点,页头页脚加上自己的站点简介等。

本文未完,请阅读原文: ■

<http://developer.51cto.com/art/201204/327050.htm>

面向对象的JavaScript基本知识指南

由于 jQuery 和 MooTools 等精心开发的库, JavaScript 已成为前端开发的基础。不过,我们要留意这些优秀库中所运用的较高级概念,这点极其重要。原因何在? 因为作为 Web 开发人员,对待学习最新的编程趋势和试图把那些趋势推向极致……



由于 jQuery 和 MooTools 等精心开发的库, JavaScript 已成为前端开发的基础。不过,我们要留意这些优秀库中所运用的较高级概念,这点极其重要。原因何在? 因为作为 Web 开发人员,对待学习最新的编程趋势和试图把那些趋势推向极致,我们必须予以一视同仁。要不然, Web 开发领域就不会出现创新。所以,我们不妨花点时间来了解 JavaScript 面向对象编程的基本知识,包括类、继承和范围。

类

在我们学习如何把类实施到代码中之前,不妨讨论一下类是什么、为什么有必要学习 / 使用类。

正如 Java 文档声明的那样:“类是用来创建一个对象的蓝图。”这蓝图就像造房子过程中所用的实际蓝图。建造人员使用蓝图来评估房子有什么样的属性,房子会有什么样的功能。类是表示对象属性的一种很方便的方式,无论这对象是房子、汽车还是人。当存在的某个对象不止一个时,类就变得特别有用。

比如说,我们不使用类来比较一下两个实际的对象。这体现了程序思考过程,而不是面向对象的思考过程。我们将描述一个名叫 Rob 的男子和一个名为 Emillee 的小女孩。我们必须假定我们对人体一无所知,因为我们没有蓝图(类)

可供使用。

Rob:

1. Rob 在身体的上部有两个椭圆形的结构,相隔几英寸。这些椭圆形结构有一个黑色背景,中间是棕色。
2. Rob 有两个与地面相对平行的结构,似乎表明了人体中最垂直的部分,这仍是身体基部的一部分。
3. Rob 有两个附属物,从另外两个附属物延伸过来。这些似乎可用来抓取物件。它们似乎比较大。
4. Rob 高度约 6 英尺。
5. Rob 无意识地吸入氧,把氧转换成二氧化碳。

Emillee:

1. Emillee 在身体的上部有两个椭圆形的结构,相隔几英寸。这些椭圆形结构有一个黑色背景,中间是蓝色。
2. Emillee 有两个与地面相对平行的结构,似乎表明了人体中最垂直的部分,这仍是身体基部的一部分。
3. Emillee 有两个附属物,从另外两个附属物延伸过来。这些似乎可用来抓取物件。它们似乎比较小。
4. Emillee 高度约 1.5 英尺。

面向对象的 JavaScript 基本知识指南大全 II

5. Emillee 无意识地吸入氧,把氧转换成二氧化碳。

单单描述一个人的 1) 眼睛、2) 肩膀、3) 双手、4) 身高和 5) 呼吸行为就有大量的工作要做。要注意: 我们不得不两次给出几乎一模一样的看法,因为我们没有蓝图可供使用。虽然描述两个人不是太费劲,但是如果我们想要描述 100 个人、1000 个人或者 100 万个人,怎么办? 肯定有一种更高效的方法来描述有着类似属性的对象: 这正是类的亮点。

我们不妨使用面向对象的理念,重新考虑前一个例子。由于我们描述的是男子和小女孩,我们知道他们都是人类。所以不妨先为人类创建一个简单的蓝图。

人类:

1. 身体的上部有两个椭圆形的结构。这些椭圆形结构有一个黑色背景,中间颜色不一样。我们称之为眼睛。

2. 有两个与地面相对平行的结构,似乎表明了人体中最垂直的部分,这仍是身体基部的一部分。我们称之为肩膀。

3. 有两个附属物,从另外两个附属物延伸过来。这些似乎可用来抓取物件。它们的大小不一样。我们称之为双手。

4. 视年龄及其他因素而定,高度不一样。我们称之为身高。

5. 无意识地吸入氧,并把氧转换成二氧化碳。我们称之为呼吸。

于是我们已声明,人类的属性是,他们有眼睛,有肩膀,有双手,有身高。我们还已声明,这些

属性可能不一样。定义了人类的蓝图后,并且声明了 Rob 和 Emillee 是人类后,我们可以将已经知道的关于人类的属性运用到 Rob 和 Emillee。

Rob 是人类。

1. Rob 有棕色的眼睛

2. Rob 有肩膀

3. Rob 有大大的双手

4. Rob 身高 6 英寸

5. Rob 会呼吸

Emillee 是人类。

1. Emillee 有蓝色的眼睛

2. Emillee 有肩膀

3. Emillee 有小小的双手

4. Emillee 身高 1.5 英尺

5. Emillee 会呼吸

我们只要明确声明 Rob 和 Emillee 是人类,就可以把与人类有关的属性和功能直接运用到 Rob 和 Emillee。这让我们可以避免重新定义身体的所有部位,同时让我们可以高效地描述这两个对象之间的重要区别。

下面是关于类及对象(名为类的实例)的几个例子,以便你明白两者之间的关系。

类 Student(学生)

◆ 属性: 年级、年龄、出生日期和学生身份标志(SSID)

◆ 功能: 计算年级平均成绩、查看缺课情况、更新操行评语

本文未完,请阅读原文: ■

<http://developer.51cto.com/art/201203/324034.htm>

编者按

3月24日, Hacker News 上发起了两个投票主题帖:“你最喜欢的编程语言”(评论数 563)和“你最讨厌的编程语言”(评论数 347)。参与投票的程序员很多,引发的评论也非常激烈。

最喜欢与最讨厌的编程语言

3月24日, Hacker News 上发起了两个投票主题帖:“你最喜欢的编程语言”(评论数 563)和“你最讨厌的编程语言”(评论数 347)。参与投票的程序员很多,引发的评论也非常激烈(这个可从原帖的评论数就可以看出来)。

@程序员的那些事 于24日晚在新浪微博发起了第一个同名投票帖:“你最喜欢的编程语言”(2,170人参与投票)。于26日晚再发布了另外一个投票帖:“你最讨厌的编程语言”(1,595人参与投票)。

微博投票结果

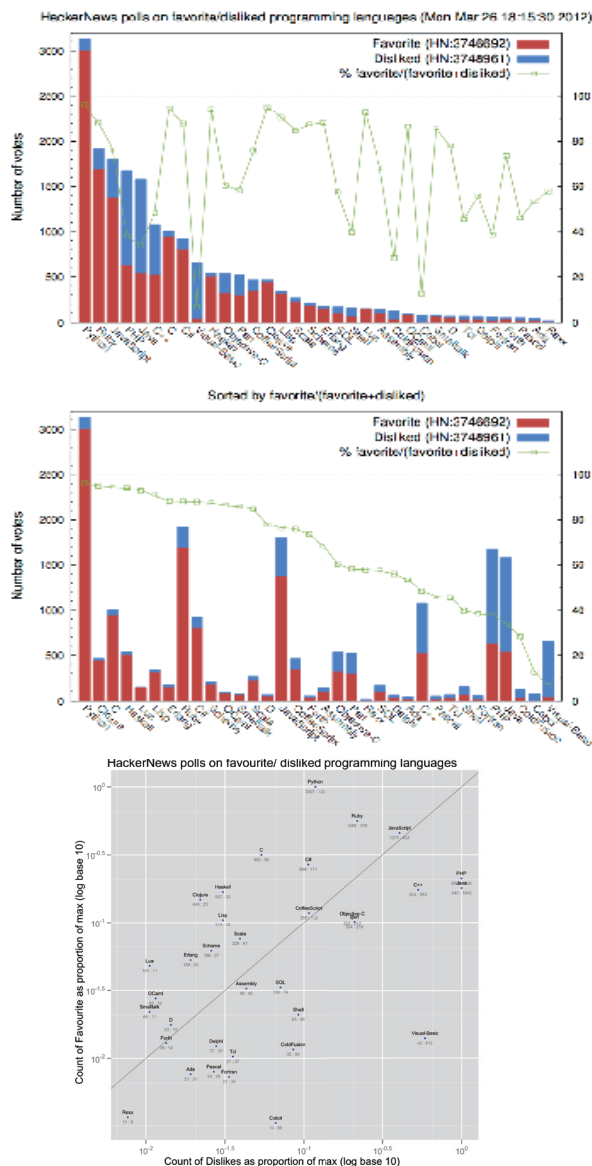
因为新浪微博投票应用最多只能有20个选项,而 HN 那两个的投票帖的备选项皆远远超过20,故而在发起投票时只选取 HN 投票中得票数已过百的选项。所以有些编程语言(比如 D 语言、ColdFusion、Forth 等)的使用者,不管是喜欢,还是讨厌,只能投给“其他”选项了。

HN 数据统计图表

HN 的那个两个投票主题帖, attractivechaos 写了一个程序,做了一张数据图。(程序代码在 Github 上)

关于上面 attractivechaos 的这张统计图,在 Reddit 有个讨论帖,有 715 个评论。

Robsteranium 不认同用柱状图,认为散点图更合适,于是自己也写了程序,绘制了一张散点图。(图片尺寸:2100 x 2100,请点击查看大图)



后话

在微博投票中, @程序员的那些事 没有鼓励大家说选择某项的理由。

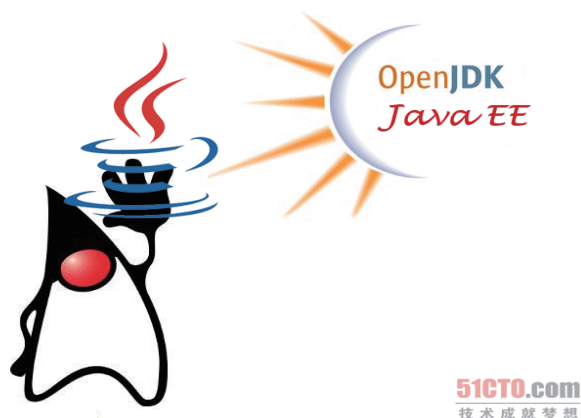
本文未完,请阅读原文: ■

<http://developer.51cto.com/art/201203/326424.htm>

Java优势依然 甲骨文更爆十年规划

前些日子“Java 已死”成为了人们的口水战,而挑起这场议论的人我并不认为他是无中生有,只是以目前语言的爆发式出现,竞争越来越激烈,人们更希望能够有一种更酷,更便捷的开发去代替它。但事实我们必须接受,时至今日,Java 已经走过了它辉煌的十几年……

前些日子“Java 已死”成为了人们的口水战,而挑起这场议论的人我并不认为他是无中生有,只是以目前语言的爆发式出现,竞争越来越激烈,人们更希望能够有一种更酷,更便捷的开发去代替它。但事实我们必须接受,时至今日,Java 已经走过了它辉煌的十几年,在这十几年来,Java 风靡全球,跃居世界开发语言榜首,更在众多企业和开发者中已经根深蒂固。而如今云计算和移动互联网的热度空前,更是带动了 Java 开发者的高度追捧。



(51CTO 配图)

拥有强大的开发者群体

目前,Java 开发者的数量已超过了之前的 900 万,将近 97% 的企业电脑也在运行着 Java,其下载量每年达到了 10 亿,如此一个庞大的数字怎么也不可能和 Java 已死联系在一起。当然,也许是 Java 的包袱太重,更多的开发者希望 Java

能够想当初那样轻装上阵,完成类似 Ruby 或 Python 的动态特性,一个简单且完备的框架及时出现。这只是希望,但并不放弃。从学习一门语言到精通,没有一个开发者会轻易的放弃。

企业级开发独领风骚

Java EE 从小型 Web 应用渗透到大型和多层次企业应用当中,现在已成为企业开发的头号语言,基于 Java 平台的应用服务器、企业服务总线、消息中间件、流程引擎这些企业应用的关键运行平台还会在相当长的时间内被广泛使用。虽然有着很多的竞争者,但却无法动摇其根本,正所谓冰冻三尺非一日之寒。Java 在拥有 97% 的企业桌面使用的同时,也不断的在企业开发上做了进一步的简易,更具备了显著优势和广阔前景。从 Java EE 7 平台进军云计算的设计方向看来,满足了平台服务提供商和应用开发者的需求,充分感受其在扩展性、弹性、多用户共享方面的优势,在云计算下更是一枝独秀。

Open JDK 开源项目

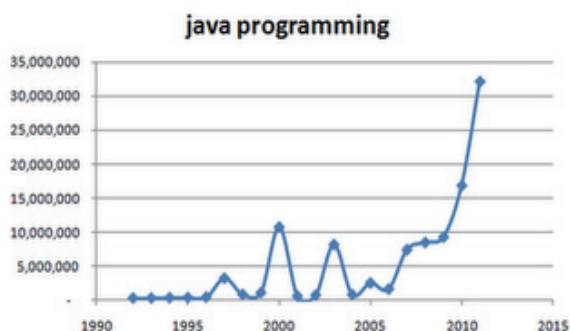
Open JDK 是实现基于 Java 平台的开源项目,连 IBM 这样的公司也纷纷加入,这让很多人都感到很惊讶。但无论怎么样,从 Oracle 和 IBM 的联盟共同努力推进 Open JDK 项目来看

Java 优势依然 甲骨文更爆十年规划 II

对于 Java 技术的未来这绝对是一个好消息。Open JDK 站在两个巨人的肩膀上不断的加速向前发展,使更多的工程师展开合作和资源共享,对于 Java 技术来说这是胜利的。

语言排行榜的优势

从今年的开发语言排行榜来看 Java 是属于下滑的趋势的,但小编认为这很正常。十几年来 Java 屹立在多如繁星的编程语言中是因为有着自己独特风格。但技术是不断在进步的,随着各种移动平台不断的受到开发者的追捧,更多类似 Java 语言的出现,其锋芒也在慢慢消退。但消退并不代表消失,就算在未来的排名中 Java 失去了王者的位置,但是 Java 的热度依然存在。以本月 Java 占据的份额为 17.110% 来说仅仅和 C 并列,并且远远超过其他语言的份额。



"Java Programmin" 关键词的搜索点击量

移动平台的普及

如今智能手机重要性的日益攀升直接带动了各种平台语言的流行,而移动平台也成为三国鼎立的趋势。在三大移动平台当中,WP 属于新兴的操作系统目前还是不能和如日中天的 iOS 和 Android 抗衡,而在这两个系统当中我认为更多的厂商和开发者还会首选 Android,原因有两个,第一;Java 在语言排行榜上的优势。第二;Android 是一个开放的平台,可供定制个性化的系统。当

然 iOS 和 WP 也有它们的优势,这里我们不讨论最后的赢家,没有竞争也不会带来发展,我只是认为 Java 目前依然是主流的开发语言。

Java 十年发展路线

从上面来看 Java 的优势显而易见,但是从长远的打算来看 Oracle 有没有为巩固 Java 做长期的计划呢? 现在就让我们来看看近日 Oracle 爆出的 Java 十年发展路线图吧。

1、JVM: 近年将开始支持越来越多的流行编程语言,包括 Groovy、JRudy、Jython、Clojure、Clojure、Kotlin、Rhino、Ceylon 等。

2、Java 8: 提高释放对象能力,易用性以及云计算方面的优化,提供统一的类型系统和优化数据结构。在云计算方面,JVM 为安全的运行多个程序提供多租户功能,及每个线程/线程组之间的资源处理和管理。

3、Java 9: 将新增一些语言上的特性,包括大数据的支持、64 位的大数组备份。异种计算模型新增了 Java 语言对 GPU 的支持、FPGA、离线引擎、远程 PL/SQL。

4、Java 10 及以上: 支持扩展函数类型,具有真正的泛型函数类型,包括多维数组数据结构优化。

5、Java 版本十年内发布时间表。详细了解: Oracle lays out long-range Java intentions

版本	发布时间
Java 8	2013 年
Java 9	2015 年
Java 10	2017 年
Java 11	2019 年
Java 12	2021 年

Java 不会死,起码在未来的 10 年内! ■

开发者应该关注的五项Web新兴技术

作者 /webapptrend

最近一位 HTML5 专家 Rich Clark (作者的好朋友) 为大家做了一个 HTML5 APIs 的简介, 在文章中为大家指向了一个令人迷惑的网页(web 平台: 浏览器技术 <http://platform.html5.org/>), 其中包含两个很长的专栏和小正文并提及到一些让人感到迷茫的技术, 例如 “window.crypto.getRandomValues” 和 “DOM Mutation observer”。

别担心, 咱们不去管那些啦, 因为有些还远远没有完成呢, 在浏览器中见到它们还要等一阵子。然而, 其它已经在浏览器中, 或者距离您很近, 或者马上就要出现。人们可能将称之为 “HTML5”, 尽管它们并不是。其实, 它们都属于令人激动的新兴 Web 技术(New Exciting Web Technology), 值得每个开发者关注。

WebGL

WebGL 是一种基于 Web 的 Graphic 库, 由非盈利组织 Khronos 运营, 结合 HTML5<canvas> 元素广泛应用在 3D 图形开发中。

学习 WebGL 比较困难, 因为它是底层开发——它运行在 GPU 上面, 而且它实际上是一个 OpenGL 的 JavaScript port, 是一种游戏开发者使用的已经长期建立的 API 集。WebGL 的主要受众是哪些已经拥有丰富 OpenGL 经验的游戏开发者, 他们可以通过 WebGL 为 web 平台编写游戏。

好在有很多资源可以帮助您学习 WebGL, 这些资源不仅仅是关于游戏开发的, 还有很多奇

幻的图形、视觉和音乐视频等方面。作者个人比较推荐的是:

◆ Introduction to WebGL。 <http://dev.opera.com/articles/view/an-introduction-to-webgl/> 作者 Luz Caballero, 简介可以获得的各种库。

◆ Raw WebGL 101。 <http://dev.opera.com/articles/view/raw-webgl-part1-getting-started/> 适合那些不使用库的用户。

◆ Learning WebGL。 <http://learningwebgl.com/> 一个非常好的引导网站。

◆ WebGL 101。 <http://www.youtube.com/watch?v=me3BviH3nZc> 一个由 Erik Moller 制作的介绍视频(2.5 小时)。

◆ See Emberwind。 <http://operasoftware.github.com/Emberwind/> 一个由 Erik Moller 做的 WebGL 游戏 port, 您可以深入 Github 或看代码。

WebGL 目前在所有桌面浏览器(发布版和开发频道) 中都支持, 除了 IE10 (微软表示不支持)。对于移动产品来说, 已经在 Opera Mobile 12 中发布了, 最终会出现在 Android, BlackBerry Playbook 2.0, Nokia N900, SonyEricsson Xpertia Android Phones 等以及 Firefox 移动浏览器中。

SVG

SVG (Scalable Vector Graphics) 已在 Opera, Firefox, Chrome 中存在多年, 但是直到 IE9 开始支持后才渐渐变得主流一些。它在 HTML5<Canvas> 下显得有点暗淡, 尽管 SVG 和 HTML5<canvas> 是面向不同应用的不同工具。

开发者应该关注的五项 Web 新兴技术 II

Canvas2D 可以迅速 paint 图形到屏幕上面,这一点很犀利。但是其全部功能就是 paint 了,没有内存来做那些(位置,顶层或其他)其他功能。如果您需要那种 book-keeping 工作,就只能自己用 JavaScript 实现,因为 Canvas2D 不会把 DOM 保存到内存中,也正因为如此 Canvas2D 速度快,十分适合第一人称射击类应用。

与 Canvas2D 不同,SVG 在您需要保存 DOM 的时候就给力了。使用 JavaScript,所有的 Objects 都可以移动并且与动画无关。您可以试试 Daniel Davis 做的复古类 SVG 游戏 Inbox Attack (<http://people.opera.com/danield/svg/inbox-attack.svg>)来体验一下,并且看看源代码来了解如何完成动画效果。

因为 shape 和 path 是用 Markup 来描述的,所以他们可以用 CSS 来定型。与 <canvas> 不同, text 在 SVG 中保持 text 格式并且更加的灵活,更加可扩展,更加易于访问。在 Canvas 中, text 变成了像素,就像 Photoshop 中的图形 text。

SVG 最强大的特性是它基于矢量,这样您的插图,图形和 UI 图标等都是矢量图了,这样无论是在 50 英寸的电视屏还是手机屏幕桌面上,看上去感觉都是一样的清晰。在当今这样一个 web 应用无处不在的时代,SVG 图形甚至可以包括媒体查询(<http://my.opera.com/ODIN/blog/2009/10/12/how-media-queries-allow-you-to-optimize-svg-icons-for-several-sizes>),可以是响应式的,可以根据不同的目标设备做尺寸的调整。

综上所述,在最新的桌面浏览器中 SVG 已经

能被广泛支持了。在移动产品方面的支持总体上来说也很好,以及预期在 Android 3.0 版本之前原生浏览器也会支持它了。

Daniel Davis 有一些 SVG 介绍性的资源 (<http://my.opera.com/tagawa/blog/learning-svg>),作者个人也推荐一本免费的电子书: Learn SVG (<http://www.learnsvg.com/book-learnsvg/>),您也可以看看《SVG or Canvas? Choosing Between the Two》(<http://dev.opera.com/articles/view/svg-or-canvas-choosing-between-the-two/>)来了解二者的区别。

getUserMedia

不像那些被错误地称为 HTML5 的 API, getUserMedia (在下文中简称为 gUM)有个相对正当的理由:起初它是 HTML5<device> 元素,之后它改名了然后离开了 W3C WebRTC 规范集合。

gUM 允许访问用户的摄像头和麦克风,本来是在 WebRTC 规范中在浏览器中进行 P2P 视频会议的,当 gUM 拥有了其他的用途,就离开了 WebRTC。

摄像头的访问最终在 Opera12 安卓版, Opera 桌面实验室和 Google Chrome Canary 里面实现了,不过 Opera 和 Chrome 都还没有实现麦克风的接入。

W3C 规范依然在用,所以 Opera 和 Webkit 有不同的语法规则,这样的麻烦被一个叫做 The gUM Shield 的小 JavaScript 片段搞定了。

本文未完,请阅读原文: ■

<http://developer.51cto.com/art/201203/324839.htm>

■ 编者按

某些语言在出现之后逐渐成为业界的骄子,而另一些则自诞生以来始终默默无闻,难以跻身主流之列。编程语言新生儿们到底因何成功,又是怎样失败的呢?

编程语言新生儿到底因何成功 又为何失败?

某些语言在出现之后逐渐成为业界的骄子,而另一些则自诞生以来始终默默无闻,难以跻身主流之列。编程语言新生儿们到底因何成功,又是怎样失败的呢?

51CTO 推荐专题:51CTO 编程语言排行榜

我们身边如今正充斥着大量不同种类的编程语言,数量之多甚至难于统计,而其中绝大部分我们都不曾耳闻。但在这里我们要讨论的是,为什么会这样?为什么某些语言获得主流认可,在业界内大红大紫,而有些则始终默默无闻?

在阅读了由 Cristina Videira Lopes 撰写的博文后,我对这一话题很感兴趣,正是她在文章中对这一话题发起分析。Lopes 指出,只有极少数成功的现代语言在学术界站稳了脚跟,而她作为学术界中的一员,对这一现状深感失望。

她的这种反应是否正常?根据 Lopes 的说法,当下几乎任何人都有能力创造出新的编程语言——情况也确实如此,从业余编程爱好者到大型企业,关注 IT 行业的每个人都在打造新语言方面跃跃欲试。与上个世纪七十年代不同,如今创建一套编程语言已经算不上什么浩大的工程。尽管如此,这些语言新生儿们大多数无法取得成功,它们最终的结果要么是成为纯粹的内部试点项目,要么是沦为实验探索好奇心的理论性产物。

为什么会这样?编程语言到底要怎样才能取得成功,那些成功的语言与默默无闻的其它语言相比又具有哪些独特的优势呢?通过对以往数款成功语言的简要调研,我们从中找到了一些端倪。

成功的语言令人感到熟悉与亲切

我们目前手头常用的数种语言已经完全能够应对工作中的各类需求。开发人员中很少有人愿意彻底放弃自己固有的知识基础,从零开始涉猎一门全新的编程语言。出于这一原因,新的语言不应该与原有主流语言之间存在过大的差异。推出一套新的标准库当然很正常,但新语言的核心语法必须令人感到熟悉与亲切。

也就是说,新语言应该至少在某种程度上类似于 C 语言。毫无疑问,这种设定会令那些讨厌 C 语言、想到大括号及其难以捉摸的执行机制就犯恶心的程序员们感到极度痛恨;然而 C 语言已经存在了 35 年了,在它身后还有一众包括 C++、Java、JavaScript 以及其它类 C 型语言的追随者,这一切都表明我们如今真的很难找出一位完全不了解 C 语言及其描述风格的程序员。

某些现代语言的确选择了不同的发展方向,Python 就是其中最典型的例子。不过 Python 所使用的独特语法并不是为了哗众取宠或是故意显得特立独行,这一切与语言自己的特点与功能息

编程语言新生儿到底因何成功 又为何失败? II

息相关。这很重要,你可以是个古怪的人,但不能为了古怪而让自己显得古怪,在这一点上大多数不成功的独特语言都未能拥有像 Python 那样的说服力。

此外,新语言应该支持大家熟悉的编程风格。创新没有问题,但最好别在大家所熟知的方面搞创新。我们可以想象,一种不从某种角度提供面向对象特性的语言,根本不可能在今天的 IT 环境之下走得很远。

当我们在自己的语言中引入全新概念时,最好抱着婴儿学步的态度。多种主流语言已经开始从函数型程序设计中汲取灵感,例如闭包以及高阶函数等内容,而这些也的确赢得了广泛赞誉。但需要强调的一点是,纯函数型语言仍然很不受欢迎。开发人员喜欢创新型理念,但绝不希望自己为了适应一款新语言而被迫从自己最熟悉的安乐窝中搬出来。

成功的语言必须开源

别再纠结于专有语言了,那些时光已然一去不复返。Visual Basic 虽然也在磕磕绊绊中一路走来,但根据我最近所做的调查,所有新型尖端语言都选择了开源——甚至由微软推出的 F# 也未能例外。

现代开发人员不喜欢那种由单独一家供应商推出、各个方面都深为掣肘的编程语言。开源给了开发人员们一种保证,令他们在当下所编写的代码不至于被未来可能出现的工具、语言或许可内容变化所影响。这种保障成果一以贯之的机制成为开发者们最为看重的优势。

但开发者层面的优势并不是开源的全部好

处,设计者们同样能从中获得种种裨益。与大多数软件不同,编译器用户群体很明显完全是由开发人员构成的。他们中的每一位成员都有能力及机会对语言及其附属工具做出改进,这一点非常重要。除此之外,这种态势还能有效帮助语言开发者对现有标准库进行扩充,这也正是 Java 大获成功的先见之明所在。

这倒不是说编程语言本身不能用来创造商业价值。免费的 C 与 C++ 编译器都已经存在了几十年,而英特尔公司也始终利用它们所带来的高性能优化编译器创造着不菲的经济收入。但这种经营模式只适用于已经步入运转正轨的语言,对于一种新型语言来说,要想在当今的市场环境中赢得广泛接受及赞赏,最基本的原则就是必须完全免费及开放。至于“专业级”市场,我们稍后再进行讨论。

利基是语言获得成功的必要前提

像 C++ 及 Java 这样的语言之所以能获得成功,是因为它们在业界整体当中找到了自己的立足点。无论大家在开发工作中偏向哪个领域,这些语言都能够为我们提供一定帮助,让自己成为我们工作当中不可或缺的一部分。

但在这种形势下,固有主流语言的地位同样难以撼动。我们已经有了 Java——真的还需要再接纳一款相似的语言吗?由于开发人员对于常用语言的依赖性相当巨大,新的通用语言几乎不可能获得与老前辈们平起平坐的竞争机会,因为利基——也就是特定需求市场已经趋近饱和。

本文未完,请阅读原文: ■

<http://developer.51cto.com/art/201203/324029.htm>